**ENVIRONMENTS**

Production environment (deployment, release environment)

DEV Environment – playground , they write requirement they copy and paste this code in

QA environment and then they test it and we send it over to production environment.

**Different type testing**

White Box testing anything related to the code, developer write code. As a QA unit testing , if there is an error you have to raise a ticket. Validating data.

Black Box testing is functional testing ,is testing the action whatever the actions are, positive negative boundary

Gray Box – data validation of data basis, front and back end. Email was sent or not backend integration

Integration testing - integration between systems , white pages servers data base testing back end, validate back end testing by making sure that the data see in the front end reflects to back end. Testing the integration of system based of data base servers or other applications.

Smoke/sanity testing- testing if the major of the system works , preparing when the basic stuff there, before you start testing. Informal testing if the fundamental features are working or nor because is not you can run testing

Functional testing

Positive scenarios black box me following the happy pot, writing test cases

Negative scenarios me against the system , writing negative test cases (invalid password)

(forget password)

Equivalence Partitioning (Boundary Testing)- how data formats are set, how are parameters set within a system, what do I need by that , for instance Facebook registration (boundary set date of birth , no older than 18, or gender, dates are set between , social security 9 digits it should let you if more or less, this is called boundary set. Parameters that you need to work with. (OVER 18 OR OVER 21)

Regression Testing is testing a system over and over again and testing entire system to make sure there is nothing wrong in the system. Moving code how it will impacted another code (possible variations .

508 compliance testing able to support people who are blind , metro cards , wants to support people, system will speak to you . basically testing are reading out .

UAT- people, business , acceptance and baseline testing for the users needs , based on them then send off, walk through, and the approving the project.

Performance testing

Performance engineer

Gorilla Testing

AD HOC – randomly testing , formal by doing functional testing, before it goes to production you play with it.

PROD ENV

QA.ENV

Dev ENV

Designed/

Built

BRD/FRD

Testing

Analysis

Deployment

Test cases, test plan , Test strategy

Regression Testing

UAT

Running Test Cases pos/neg

Report Defects

Chase bank checking example

(Matching information , data , validate, integration testing)

CHASE.COM

Saving Acc.

Checking Acc.

Saving dB

Checking dB

AGILE SCRUM

Agile – ability to move quickly and easily.

Agile is software development – under which requirements and solutions evolve through the collaborative effort . Both development and testing activities are concurrent unlike waterfall.

* Fast- paced environment
* Ever- changing environment
* Delivery focused
* Flexibility

Agile Umbrella

Methods introducing Agility

SCRUM

KANBAN- SDLS Process and the board on the wall phases, picking out certain features and delivering them

SCRUM- come from “Regby” lock hand together and pushing a project.

AGILE

incrementally

Instead of all at once

AgileVS Waterfall : Poor quality

* Analysis
* Design
* Code
* Test

In Agile you do user stories and you do analysis design, testing

Working software—sdlc you do all stuff in one time

Epic Story , Task

Not much documentation

Epic – Why? Defines the business needs?, Larger requirements Huge require.

Feature 🡪 story 🡪task .What Define the customer needs

As a user I need to use Online banking application – this is Epic Larger requirements.

Story – smaller requirements, AS a user I can login into online banking application

Story 2 As a user I cam check my balance in my account

Task is an action that we need to perform for that particular story

Developer task

Review the story

Estimate the story

Design

Code

Unit testing

Integration testing

QA

Review the user story

Creating test cases

Reviewing test cases

Setting up test environment

Creating test data

Executing manual test

Automation setup

Automation design

Automating test cases

Re- testing bugs

Regression testing

Demo user story

In Agile we have epics and stories, we get it from Product owner who is responsible to create this stories.

Agile Roles

Product owner – responsible for getting the requirements from the customer/ stakeholders

Product Backlog – prepared by Product Owner which contains requirements from the customer.

Sprint planning we need to decide what is the duration, estimating stories

Story point is an estimation given by tester and developer for the specific story